第 36 卷 第 8 期 计 算 机 工 程 2010 年 4 月
Vol.36 No. 8 Computer Engineering April 2010
·软件技术与数据库· 文章编号：1000—3428(2010)08—0076—03 文献标识码：A 中图分类号：TP391

虚拟机文件取证分析

钟 琳[1]，许榕生[2]
(1. 福州大学数学与计算机学院，福州 350000；2. 中国科学院高能物理研究所计算中心，北京 100049)

摘 要：介绍 VMware Workstation 软件常见的虚拟机磁盘文件结构及其含义，提出一种符合司法要求的虚拟机文件取证方法。通过直接扫描虚拟机文件，依据虚拟硬盘分区文件类型的存储结构定位虚拟机磁盘文件的虚拟主机的相应扇区，获取证据。以虚拟硬盘分区文件系统 NTFS 为例进行说明，探讨在虚拟机下进行计算机取证的策略。

关键词：虚拟机；VMware Workstation 软件；取证

Analysis of Virtual Machine File Forensic

ZHONG Lin[1], XU Rong-sheng[2]
(1. College of Mathematics and Computer, Fuzhou University, Fuzhou 350000;
2. Center of Computing, Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049)

【Abstract】 This paper introduces the structure and meanings of virtual machine disk file under VMware Workstation software. And the approach for the virtual machine files for judicial need is analyzed. According to the file storage structure in the partition file system, it scans the virtual machine file to locate the sector in the virtual host of the virtual machine disk file for accessing to evidence. Taking virtual hard disk partition to NTFS file system as an example, it discusses some forensic strategies under a virtual machine.

【Key words】 virtual machine; VMware Workstation software; forensic

虚拟机技术是近几年计算机领域最热门的技术之一。每个虚拟机模拟一个完整的计算机系统，以文件的形式存储在物理机中。为了保证证据的原始性，取证过程中遇到虚机文件不允许直接使用虚拟机软件加载。因此，如何有效提取该文件信息成为值得思考的问题。本文着重介绍 VMware Workstation 中 Windows 系统的取证过程。

1 虚拟机文件结构

虚拟机数据是以文件的形式存储的，研究的重点是虚拟机磁盘文件(.VMDK)。VMware Workstation 采用主机稀疏盘区(Hosted Sparse Extents)形式[1]，结构如图 1 所示。

图 1 主机稀疏盘区

1.1 粒目录、粒表、粒

粒目录(GD)、粒表(GT)、粒[1]是 VMware 虚拟磁盘文件特殊的层次结构。其中，粒目录指向粒表；粒表指向粒；粒是若干扇区块，包含虚拟磁盘信息。默认值为 128 扇区或 64 KB。粒目录由一组 32 bit 的粒目录项组成。粒表由一组 32 bit 的粒表项构成，一般大小为 2 KB。

1.2 稀疏头

VMDK 文件的第 1 个扇区叫作主机稀疏盘区头(Hosted Sparse Extents Header)[1]，以 ASCII 的“VMDK”开始，记录有关层次结构的各种参数，其所需参数见表 1。

表 1 所需稀疏头参数

偏移/bit	大小/Byte	描述
0x0B	8	磁盘容量 VMSz
0x14	8	粒大小 GSz
0x2B	4	每粒表的粒表项数 GTEsPerGT
0x38	8	粒目录的起始扇区 GDOffset
0x40	8	起始粒扇区 VMFstSec

1.3 虚拟磁盘

虚拟磁盘是粒的集合。虚拟磁盘的存储结构与同种文件系统在真实硬盘上的结构相同。因此，只需定位虚拟磁盘，根据其文件系统类型进行证据的获取。

2 虚拟机文件中的 NTFS 文件系统

2.1 主引导记录区和虚拟主引导记录区

MBR(主引导记录区)[2]共 512 Byte，最后 64 Byte 为硬盘分区表(Disk Partition Table, DPT)。偏移 0x04 的字节是该分区的文件系统类型 FileSys，偏移 0x08 起的 4 Byte 为本分区已使用扇区数 UsedSecs，其后 4 Byte 是本分区总扇区数 TotalSecs。虚拟 MBR[2]与 MBR 结构相同。

2.2 引导扇区

DBR(DOS Boot Record)[2]是操作系统引导扇区，包括引

基金项目：北京市优秀人才培养基金资助项目(20061D0500700165)；北京市教委科技发展计划基金资助项目(KM200610772006)
作者简介：钟 琳(1985－)，女，硕士研究生，主研方向：网络安全，计算机取证；许榕生，研究员、博士生导师
收稿日期：2009-09-22 **E-mail：**zhonglin@ihep.ac.cn

导程序和 BPB(BIOS Parameter Block)。本文所需的 BPB 参数见表 2。

表 2　所需 BPB 参数

偏移/bit	长度/Byte	描述
0x0B	2	每扇区字节数 BytsPerSec
0x0D	1	每簇扇区数 SecsPerClus
0x30	8	$MFT 的起始簇号 MFTFstClust

2.3　主文件表

在 NTFS 下，文件通过主文件表(Master File Table, MFT)确定其在磁盘的存储位置[2]。每个文件对应一个文件记录，MFT 是这些文件记录的集合。

文件记录从 0 开始编号。本文所需的$MFT 编号为 0；根目录(\)编号为 5，保存所有文件和目录的索引；位图文件($Bitmap)编号为 6，用于判断文件的完整性。

每个文件都存在唯一的 64 位文件引用号。文件引用号包括文件号和文件顺序号。文件号是低 48 位，即上文的编号，定位文件在 MFT 中的位置。

2.4　MFT 文件记录

MFT 文件记录大小一般是 1 KB。文件记录由头部和属性流构成。以 ASCII 的“ FILE ”开始。文件记录头部偏移 0x16 的标志字节，高位为删除标志，低位表示文件/目录，因此，00H 表示删除的文件；01H 为正常的文件；02H 为删除的目录；03H 为正常的目录。

2.4.1　常驻属性和非常驻属性

NTFS 将文件作为属性/属性值的集合处理。若属性值直接存放在文件记录中，称为常驻属性。非常驻属性[2](如数据流属性)因属性值太大而不能存放在 MFT 文件记录。NTFS 将从 MFT 之外的位置分配空间，这些空间称为运行(Run)。

2.4.2　文件名属性

文件名属性是常驻的，类型为 0x30。前 8 Byte 为父目录的文件引用号。其后的 4 个 8 Byte 分别为创建时间、修改时间、最后修改时间、最后访问时间。偏移 0x41 的字节为文件名命名空间。命名空间是 NTFS 为了兼容 FAT 文件系统 8.3 命名空间而设置的。数据恢复只需长文件名，因此，只提取命名空间为 01H 和 03H 的文件名。

2.4.3　数据流属性

数据流属性的特征为 0x80。偏移 0x08 的非常驻标志字节判断是否常驻。为 0，表示常驻；为 1，说明非常驻。Slack 恢复所需偏移 0x28 的 8 Byte 的分配空间 AllocSize，其后的 8 Byte 的实际空间 RealSize 都以 Byte 为单位。

2.4.4　索引根属性和索引分配属性

索引根属性是常驻的，特征为 0x90。索引分配属性是非常驻的，特征为 0xA0，是否存在由索引根属性偏移 0x0C 的标志字节决定：为 1，表示存在，否则不存在。索引分配是一组运行、定位其他索引的位置。

2.5　索引记录

NTFS 的目录一般存在 MFT 的索引根属性中，若过大则存放在索引分配属性指向的索引缓冲区。

索引记录包括索引头和一组索引项，一般为 4 KB。

2.5.1　索引头

索引文件是以 ASCII 码“ INDX ”开始的，偏移 0x24 的标志字节为 1 表示有子节点；为 0 表示是叶节点。

2.5.2　索引项

偏移 0x10 起的 8 Byte 为父目录的文件引用号。其后的 4 个 8 Byte 参数分别为创建时间、最后修改时间，最后访问时间以及子目录或文件的名称。

3　虚拟机文件取证过程

3.1　虚拟磁盘的定位

虚拟磁盘定位的具体方法(见图 2)如下：

(1)定位 VMDK 文件的起始扇区，获得该文件起始扇区的句柄。

(2)定位到虚拟机系统的起始扇区，读取主机稀疏盘区头，将句柄后移 VMFstSec 个扇区。

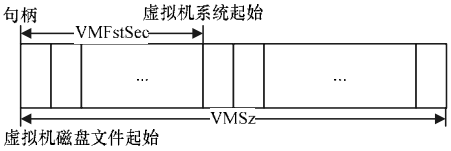

图 2　虚拟磁盘的定位

3.2　虚拟磁盘扇区的定位

定位虚拟磁盘的扇区 x 方法如下：

(1)定位到粒目录的起始扇区(FstGD)，即将虚拟磁盘起始扇区(FstVM)句柄后移 GDOffset 个扇区。

(2)查询对应的粒目录项，获取粒表的扇区偏移量。GT=GDE[floor(x/GTCov)]，floor 是下界函数，粒表可访问扇区数，GTCov=GSx×GTEsPerGT。

(3)定位到该扇区对应的粒表所在的扇区 GTESec=GT+FstVM。

(4)查询对应的粒表项，获得在虚拟磁盘的扇区偏移量 GTE=GT[(floor(x%GTCov)/GTEsPerGT)]。

(5)定位虚拟磁盘扇区 VMSec=GTE+x%GTCov+FstVM。

3.3　Windows 分区的定位

如图 3 所示，读取 MBR 或虚拟 MBR，获取 DPT 中 FileSys, UsedSecs, TotalSecs 等参数值。按照以下分区定位公式，获得各分区起始位置：

当前分区的起始扇区 = MBR(虚拟MBR)的起始扇区 + UsedSecs

下一个虚拟MBR起始扇区 = 当前分区起始扇区 + TotalSecs

MBR 的起始扇区为 0。

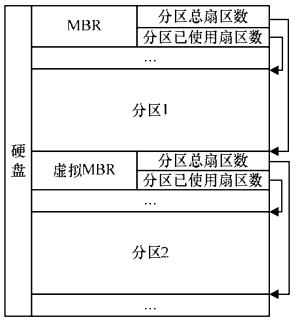

图 3　Windows 分区的定位

3.4　NTFS 下的文件定位

本文以 Windows 主流文件系统 NTFS 为例。为了保证文件的原始性，计算机取证的每一步操作都不允许调用操作系统自带的文件操作命令，而需直接扫描硬盘，按文件系统的存储结构定位到文件的位置。NTFS 文件定位步骤如下：

(1)定位到特定分区，获取一个句柄。

(2)在 DBR 中获取 BPB 表的所需参数，定位到 MFT，即移动句柄到该分区 MFT 的起始扇区

$$FstMFTSec=FstClustOfMFT\times SecsPerClust$$

(3)读取根目录的文件记录信息，定位到根目录，即句柄后移 10 个扇区。

(4)对于文件，根据文件参考号定位文件的文件记录，读取文件记录信息，获取数据流属性，若常驻，直接读取，否则，读取数据运行列表，定位数据簇。

(5)对于文件夹，若该目录的下一层文件(夹)不多，直接从该目录的文件记录中获取信息，反之，必须定位该目录的索引记录，再获取相关信息。

MFT 区域的定位是 NTFS 文件系统下获取文件痕迹的基础。因此，首先需要定位 MFT 区域。

3.5 NTFS 下证据获取方法

计算机取证的最重要目的是了解什么时间发生了什么事情。因此，一个必不可少的证据是时间属性。NTFS 的时间属性有 4 项，分别为文件创建时间、文件修改时间、最后修改时间、最后访问时间。这些时间属性包含在文件记录的文件名属性中，也存放在索引记录的索引项中。

3.5.1 已删除文件(夹)

获取已删除文件信息就是收集文件记录未被覆盖文件的相关信息。一般做法为：逐一扫描 MFT 文件记录，检查文件记录头中的标志字节是否为 00H。若是，分析文件名和数据流等属性，记录文件名、数据运行列表、父目录文件引用号等相关信息。之后，根据父目录文件引用号重建删除前目录关系。

已删除文件的获取还可作为数据恢复的前期工作，即根据获得的数据运行列表计算该文件占用的簇，然后查找位图文件的相应位：若都为 0，则表示文件完整，即可恢复出完整的文件；否则，表示文件已破坏，获得文件碎片，进行下一步工作。

文件删除后，所属的文件夹也要删除文件的索引项。若是非常驻的，则索引记录后面的数据前移[3]。因此，通常只对文件记录搜索，不涉及索引记录。但如果删除的是文件夹，系统回收数据簇，只要未再分配，就不会改变索引数据簇。虽然这样不一定能定位和恢复文件，但是可得到已删除目录和更完整的已删除文件信息。具体实现方法为：在扫描 MFT 的同时，判断文件记录头标志字节是否为 02H：若是，分析索引根属性，若存在索引分配属性，则定位索引数据簇。获得该目录下文件名或目录名等信息。之后重建删除前目录关系，见图 4。这样，使用了所有的存储记录，可得到更完整详细的已删除文件信息。

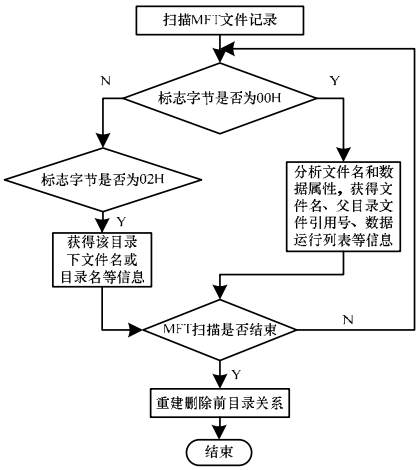

图 4 已删除文件(夹)恢复流程

3.5.2 文件残存

文件残存(File Slack)是指硬盘中某些隐藏重要数据的未使用空间。一般处于文件末端与文件最后一簇的末端之间。File Slack 分为 2 种：一种是 RAM Slack，指文件末端到该扇区末端的区域，另一种称为 Drive Slack，即文件末端的下一个扇区到最后一簇的区域，如图 5 所示。

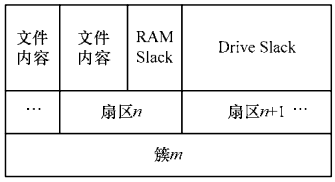

图 5 文件残存类型

定位文件的文件记录，获得 AllocSize 和 RealSize，然后从文件末端起定位。

FileSlack=AllocSize-RealSize

DriveSlack=int(FileSlack / BytsPerSec)×BytPerSec

RAMSlack=FileSlack-DriveSlack

3.5.3 部分初始化文件

部分初始化文件(Partial initial file)是指初始化时文件大小为固定值 Default Size 的文件。常见的有系统日志、CRMlog 文件(注册表文件)、Outlook pst 文件、数据库等。

各种文件的查找方式如下：

(1)系统日志：包括应用程序日志、安全日志、系统日志、DNS 日志，它们的默认位置：%systemroot%/system32/config。默认文件大小为 512 KB。其中，

安全日志文件：

%systemroot%/system32/config/SecEvent.evt

系统日志文件：

%systemroot%/system32/config/SysEvent.evt

应用程序日志文件：

%systemroot%/system32/config/AppEvent.evt

(2)CRMLog 文件存放于%systemroot%/Registration，固定大小为 1 024 KB。

(3)Outlook pst 文件：可存放在系统的任意位置，因此，直接搜索文件扩展名。默认设置为 265 KB。

图 6 为 WinHex 查看 AppEvent.evt 文件记录，其中可得 AllocationSize=0x10000, RealSize=0。

```
80 5C 4E 48 00 00 00 00  30 00 00 00 78 00 00 00  €
00 00 00 00 00 00 02 00  5A 00 00 00 18 00 01 00  .
1E 00 00 00 00 00 01 00  20 B2 4C E8 51 3B C9 01  .
20 B2 4C E8 51 3B C9 01  20 B2 4C E8 51 3B C9 01
20 B2 4C E8 51 3B C9 01  00 00 01 00 00 00 00 00
00 00 00 00 00 00 00 00  20 00 00 00 00 00 00 00  .
```

图 6 AppEvent.evt 的文件记录

若按照上节的算法，说明整个文件都属于 File Slack。但是使用事件查看器查看该文件信息，不难发现文件中包含大量信息，如图 7 所示。因此，这类文件需特殊处理。

图 7 AppEvent.evt 属性

(下转第 82 页)